
Text Conditional Lyric Video Generation

Nicholas Frosst
Google, Good Kid
frosst@google.com

Jonathon Kereliuk
Good Kid
jonathon@goodkidofficial.com

Abstract

The artistic applications of generative adversarial networks (GANs) has been established for some time now. Recent work on text conditioned image generation has opened up a new avenue of GAN artistry. By making use of AttnGAN, conditioning on the lyrics of a piece of music and generating images that smoothly interpolate between the lyrics in time with their delivery, we are able to generate interesting and engaging lyric videos. In this paper we propose this simple technique and explore some modifications which make the results more visually appealing.

1 Background

Generative adversarial networks (GANs) were first introduced by Goodfellow et al. [2014]. Since their introduction, many researchers have expanded on the idea of using an adversarial trained discriminator and noise conditioned generator to create realistic samples. Recent work of Xu et al. [2018] made use of an attentional mechanism to create natural language conditioned image generators. With their method, and making use of the MSCOCO dataset [Lin et al., 2014] they were able to create convincing images that resembled natural language descriptions. This has opened up the possibility of using text conditioned image generation for artistic purposes.

2 Technique

By making use of the text conditional GAN network AttnGAN, we are able to generate an image for every lyric of a song. We simply use each individual line in the song as the input for the model, thereby creating a series of anchor images which represent the lyrics. In order to turn this series of images into a video we interpolate between these anchor images in time with the music. To do this we first need to have a time-stamped version of the lyrics, in which each individual line is annotated with the time at which the line is sung. One could conceive of an algorithmic way of doing this, but in our work we made use of a python tool which allows a user to listen to a song and read along with the lyrics, the user then presses a button when each new line is sung. In this way we created time annotated lyrics that allowed us to calculate the duration of each lyrical transition. For each lyrical transition we calculated the number of intermediate images that would be required to maintain a consistent frame rate. We then calculated the representation space for each of these transitional images by interpolating between the representation space of the two anchor images, conditioned on the neighboring lyrics. In this way we were able to create videos with consistent frame rates that seem to dance from one lyric to the next.

Specifically for each lyrical in song, we create a corresponding anchor images by sampling from the AttnGAN conditioned on that lyric. We save the representation space vector derived from each lyric as l_i . To calculate an intermediate frame $\theta \in [0, 1]$ between two anchor frames i and j , we calculate the representation vector $l_{(i,j,\theta)} = l_i$ as the linear interpolation between l_i and l_j .

$$l_{(i,j,\theta)} = l_i + \theta(l_j - l_i) \quad (1)$$

and generated an image from the AttnGAN conditioned on this representation space. We calculated enough intermediate frames to fill the time between all the lyrics. We found that the effect was slightly more eye catching when there was a rapid change of the generated images that occurred in time with the delivery of a new lyric. To create this effect we cut the interpolation between the two anchor images short, meaning that there would be a large visual change when a new lyric was delivered. Specifically calculated the intermediate representation vectors $l_{(i,j,\theta)} = l_i$ as

$$l_{(i,j,\theta)} = l_i + 0.9(\theta)(l_i - l_j) \tag{2}$$

The AttnGAN architecture splits up the representation space into two sections, a global sentence vector (representing the sentence as a whole) and matrix of word vectors (representing the individual words in the sentence). This allows the model to attend to individual words as well as the full sentence. When interpolating between the anchor images we found that concatenating these vectors together into a single vector and interpolating between these concatenations yielded the best results when compared to using just the sentence vector.




	Anchor frame 1	Intermediate frame	Anchor frame 2
AttnGAN input	"We got here in the dark, we took the long way down."	Representation space linear interpolation between frame 1 and frame 2	"Just to find an in somewhere inside this ancient town."
AttnGAN output			

Figure 1: Examples frames from video generated for Witches by Good Kid [Kozakov et al.]. The two images for anchor frame 1 and 2 represent images sampled from the GAN by conditioning on the respective lyrics. The intermediate frame was created by linear interpolating between the representation space of the anchor frames.

3 Data Distribution

We used a model trained on the MSCOCO dataset which is comprised of natural images and English language descriptions of those images written by people. As such the types of English sentences that the model was trained on are quite a bit different from the lyrics presented in English pop rock songs. The sentences in MSCOCO are all very concrete and descriptive, while lyrics are often vague and poetic. Qualitatively it was apparent that lyrics with many concrete descriptive words resulted in better looking videos. This distributional shift could be corrected by collecting a new dataset in which the creators were explicitly tasked with describing the scene with poetry. Alternatively one can imagine a technique for extracting the content words from a lyric and crafting a descriptive sentence with them, and using this as an input to the model. For our work we found the failure modes that we encountered when the input was particularly poetic enjoyable and so no attempt to correct this was made, but this provides an avenue for future artistic work.

4 Spherical Linear Interpolation

Previous work on GAN interpolation by White [2016] has shown that a linear interpolation yields worse results than spherical linear interpolation. It was suggested that by keeping the length of the representation vector fixed, the model created better samples. When using the attention GAN we found the difference between these two methods negligible and so we made use of a linear interpolation, for simplicity.

5 Conclusion

We have presented a simple technique for making neural net generated lyric videos. by using a text conditional image GAN and conditioning on the lyrics, we are able to create a representative anchor frame for each lyric. We then smoothly interpolate between these images in time with the lyrical delivery in order to create a visually appealing video which represents the lyrics.

Ethics

When making use of a generative model for artistic purposes one cannot be entirely be sure of what images will be created. In our work we made use of a pretrained model trained on a publicly available dataset. As a result not only were we unsure of what types of images we were going to generate, we were not entirely aware of the data set on which the network was trained. Because GANS have been known to overfit to the training data, it is very much possible that the likeness of a person in the dataset would be generated in part of our video. In this way it could be possible for someone to have their image displayed unknowingly in a video released online. We have examined the videos that we generated and no particular likeness is displayed, let alone one of a particular individual, but as generative art improves this may be an issue worth considering.

Acknowledgments

We would like to thank the other members of our band David Wood, Jacob Tsafatinos and Michael Kozakov, for creative feedback and use of music, as well as Pablo Castro for feedback and suggestions.

References

- Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.
- Micheal Kozakov, David Wood, Nick Frosst, Jacob Tsafatinos, and Jonathon Kereliuk. URL <https://open.spotify.com/album/1K99wKox6kriQyCMJCAsfD>.
- Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.
- Tom White. Sampling generative networks. *arXiv preprint arXiv:1609.04468*, 2016.
- Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1316–1324, 2018.